

AMoRe
a package for
A-utomatic *M*o-lecular *R*_ε-placement

Tutorial

Jorge Navaza¹
C.N.R.S. France

¹Acta Crystallographica A50, 157-163. (1994) “*AMoRe* : an Automated Package for Molecular Replacement”.

AMORE is a suite of programs aimed at locating model electron densities within the crystal cell.

Contents

The package contains:

factor.d	=	ascii file of atomic form-factors.	
symop.d	=	ascii file of space-group symmetry operations.	
maitre.for	=	program and subroutines to be modified, if necessary, by the user.	
setup	=	script to initiate a molecular replacement problem.	
tutorial.ps	=	postscript file with tutorial.	
writeup.ps	=	postscript file with manual.	
BIN_sgi_5.3_f77	=	configuration dependent sub-directories containing the library of compiled subroutines and executables	eslave.a
BIN_sgi_6.2_f90			amore.exe
BIN_...			cording.exe
BIN_alf_4.0_f90			splits.exe

The user must define the environment variable **AMORE** equal to the path-name of the directory where the **AMORE** package is, and the environment variable **BIN** equal to the name of the sub-directory (within \$AMORE) where the appropriate binary files are (archive and executables). These binary files are machine/OS-version/compiler dependent; each configuration has its own particular sub-directory.

By default, the programs use the executables in \$AMORE/\$BIN, but the user has the possibility of producing its own executables. For this, he needs to define the appropriate environment variable **COMPILE** (compiler options as used to generate eslave.a in the corresponding sub-directory \$BIN),

Example of .login file (for csh)

```
setenv AMORE "{pathname-of-AMOR $\epsilon$ }"

switch ('hostname -s')

case 'sgi5':
# Silicon Graphics Indigo under OS 5.3 and compiler f77
setenv BIN "BIN_sgi_5.3_f77"
setenv COMPILE "f77"
breaksw

case 'sgi6':
# Silicon Graphics Indigo under OS 6.5 and compiler f90
setenv BIN "BIN_sgi_6.5_f90"
setenv COMPILE "f90"
breaksw

case 'alpha4':
# Alpha DEC under OSF1 V4.0 and compiler f90
setenv BIN "BIN_alf_4.0_f90"
setenv COMPILE "f90"
breaksw

...

endsw
```

Programs

Preliminary programs to cast input data into a suitable representation:

SORTING	packs and sorts H,K,L, F^{obs} . The packing is cell and space-group dependent.
TABLING	calculates the array of molecular scattering factors corresponding to the model coordinates (or electron densities). Puts the model in a small model-box. First translates the coordinates so that the center of mass – CoM – is at the origin, and rotates the coordinates so that the model's principal axes of inertia are parallel to the model-box axes. The model-box is put in a big cell in order to sample the model transform finely, to allow structure factors and gradients of the rotating model to be accurately interpolated from the array.

Main molecular replacement programs:

ROTING	calculates spherical-harmonics expansions of crystal and model Patterson functions and computes several rotation functions.
TRAINING	computes n-body fast translation functions. The output is, for each orientation, the correlation coefficients and R-factors of the top peaks of fast translation functions.
FITING	performs least-squares fast rigid-body refinement.

Auxiliary programs:

PATTING	calculates the Patterson function.
JOB	creates a starting set of inputs and a script with a tentative protocol to solve the molecular replacement problem.
OIC	prepares inputs to main molecular replacement programs by selecting and combining intermediate results.
MR2IC	works out the final rotations and translations to apply to the initial models.
CORDING	produces coordinates corresponding to the best solution.

Automation

The package may be used at three different levels of automation:

- LEVEL-3: a starting set of inputs and a script with a tentative protocol to solve the molecular replacement problem, are created by program JOB. The input is general purpose control parameters (e.g. translation function options, number of peaks to select).
- LEVEL-2: inputs to the main molecular replacement programs are created by program OIC, which selects and combines available potential solutions. Information produced in previous steps may be used in order to create efficient inputs for OIC (e.g. cutoff values to skip potential solutions). The procedures `oic_rotting`, `oic_traing` and `oic_fiting` are the concatenations of OIC with the main molecular replacement programs.
- LEVEL-1: inputs to the main programs are prepared by the user. In general, they need information produced in previous steps (e.g. positions to be refined).

Starting a Molecular Replacement Problem

To start a molecular replacement problem, it is recommended to move to an empty working-directory and execute

- `cd $AMORE/setup`

An alias may be defined to perform it. The setup procedure creates sub-directories, and puts files into them. The initial content of the working-directory and sub-directories is:

./	aide-memoire	=	succinct notice.
	cording	=	script to generate output coordinates.
	splits	=	script to split models.
	./d/		= sub-directories.
	./e/		
	./f/		
	./i/		
	./o/		
./d/	data.d	=	example of main AMORE input.
	hkl.example		= examples of diffraction data and coordinates files needed by AMORE .
	xyz1.example		
./e/	maitre.for	=	program and subroutines to fit data and memory requirements (copied from \$AMORE).
	makeupd	=	script to make a new (local) executable, by compiling maitre.for and linking with \$AMORE/\$BIN/esclave.a .
	entorno		= program commands.
	job		
	sorting		
	patting		
	tabling		
	roting		
	traing		
	fiting		
	oic		
	oic_roting		
	oic_traing		
	oic_fiting		
	mr2ic		
./f/	empty	;	it will contain binary files created by programs.
./i/	dato.i3	=	example of LEVEL-3 input to run AMORE automatically.
		;	it will contain inputs to programs; names with prescribed syntax (e.g. dato.i3 , sort.i1 , ...).
./o/	empty	;	it will contain outputs of programs, named {\$}.s .

Move to:

- `./d/`

adapt the data.d file. The symmetry cards must be lower case, finishing '`* end`' or '`* stop`'; no more than 80 characters per line; first equivalent position must be '`x,y,z`'. Centering translation may be explicitly introduced (see Example below and file \$AMORE/symop.d).

Files with diffraction data and model coordinates must be named hkl.d (as hkl.example) and xyz{#}.d for model number{#} (as xyz1.example), respectively. Insert the FORMAT card in upper case, in hkl.d and xyz{#}.d files.

When model electron densities are used instead of model coordinates, create the map{#}.d files, as explained in the write-up.

It is not necessary to rename files; just create symbolic links:

```
ln -sf FAB.LYS.HKL hkl.d
```

```
ln -sf FAB.PDB xyz1.d
```

```
ln -sf LYS.PDB xyz2.d
```

- `./i/`

Adapt dato.i3 , if necessary.

Programs are executed from the working-directory. Execute

- `csh ./e/job dato`

JOB creates default input cards (in `./i/`) and a script (`./job`) with a tentative protocol to solve the molecular replacement problem. The script `./job` is in fact a series of procedures which may be executed separately, as illustrated in the Example below.

To run **AMoRE** automatically, submit

- `csh ./job >& log &` (or `csh ./job`)

Example

The data correspond to a crystal with two molecules of the complex Fab-Lysozyme in the asymmetric unit, and two search models (xyz1.d = Fab, xyz2.d = Lysozyme). The input needed by **AMOR ϵ** is:

Input:

file ./d/hkl.d

The FORMAT card should allow reading [H, K, L, F^{obs}] – four items. All header cards are ignored, till the FORMAT card (format A5,*) is found. If phases are available (to use in 'p-t-f' phased translation), insert PHASE card (format A5) after FORMAT, which should now allow reading five items.

remark: this is a fab + lysozyme complex. it may have two molecules.

remark: therefore it may have two fab and two lysozyme.

remark: coordinates sent fab d1.3 hel, residues 601 up are lysozyme.

remark: the following format may not correspond to the data.

FORMAT (3I4,F12.2)

0 0 6 894.10 38.40

0 0 8 507.50 15.70

0 0 10 116.10 9.70

...

file ./d/xyz1.d

The FORMAT card should allow reading [ATOM-type, X, Y, Z, B-factor] – 5 items. If fractionary coordinates, insert CELL card (format A4) after FORMAT (see subroutine lecatc in file ./e/maitre.for).

remark: FAB FRAGMENT (ANTI-LYSOZYME ANTIBODY D1.3)
remark: the following format (typical of PDB) may not correspond to
the data.

FORMAT (12X,A4,14X,3F8.3,6X,F6.2)

ATOM	1	N	ASP	1	4.060	7.307	5.186	1.00	51.58
------	---	---	-----	---	-------	-------	-------	------	-------

ATOM	2	CA	ASP	1	4.042	7.776	6.553	1.00	48.05
------	---	----	-----	---	-------	-------	-------	------	-------

...

file ./d/xyz2.d

remark: LYSOZYME

remark: the following format (typical of PDB) may not correspond to
the data.

FORMAT (12X,A4,14X,3F8.3,6X,F6.2)

ATOM	3314	N	LYS	1	0.829	-12.809	39.408	1.00	35.17
------	------	---	-----	---	-------	---------	--------	------	-------

ATOM	3315	CA	LYS	1	0.556	-13.062	38.012	1.00	33.11
------	------	----	-----	---	-------	---------	--------	------	-------

...

Note that atom types are read with format A4. However, only the first two characters (Mendeleev notation) are used to get the atomic scattering factors from file \$AMORE/factor.d (see subroutine lecatc and lgauss in ./e/maitre.for). This may need shifting of some names. For example, Calcium, Carbon, Zinc and Oxygen should appear as:

ATOM	311	Ca	EXA	1	0.545	-10.000	38.000	1.00	30.01
------	-----	----	-----	---	-------	---------	--------	------	-------

ATOM	312	C	EXA	1	0.556	-13.062	38.012	1.00	33.11
------	-----	---	-----	---	-------	---------	--------	------	-------

ATOM	313	Zn	EXA	1	0.829	-12.809	39.408	1.00	35.17
------	-----	----	-----	---	-------	---------	--------	------	-------

ATOM	314	O	EXA	1	0.655	-11.030	39.020	1.00	36.20
------	-----	---	-----	---	-------	---------	--------	------	-------

file ./d/data.d

* D44HEL **

99.7 167.3 84.7 90. 90. 90.

x,y,z * $1/2+x, 1/2-y, -z$ * $1/2-x, -y, 1/2+z$ * $-x, 1/2+y, 1/2-z$ * end

0

95. 0.

15. 3.5

2 2 2

Description

- 1) Title (format A80).
 - 2) Cell.
 - 3) Symmetry operations (lower case), finishing '* end'; first equivalent position must be 'x,y,z'; no more than 80 characters per line. Centering translation may be explicitly introduced (see file \$AMORE/symop.d); for example, the following are both licit inputs (though the latter is recommended) for space group C2:
x,y,z * -x,y,-z * x+1/2,y+1/2,z * -x+1/2,y+1/2,-z * end
or
x,y,z * -x,y,-z * 1/2,1/2,0 * end
 - 4) NORT
Code to define an orthogonal reference frame.
 - 5) PERC BADD
PERC = uses only the PERC % highest F^{obs} within the selected resolution range.
BADD = B-factor added to F^{obs} (e.g. -5 to sharpen data).
 - 6) DMAX DMIN
Resolution limits used for the molecular replacement problem (in Å).
 - 7) NTYP MOL1 ... MOLn
NTYP = number of different models.
MOL{#} = number of molecules of model-type {#} in the a.u.;
{#} = 1,NTYP.
-

file ./i/dato.i3

job +*+*+*+*+*+*+*+*+*
xyz
1. 2 10 0.5 2.5
c-o 50 0.3 30
p-t 10 0.5 30
10 20
20.

Description

- 1) Keyword (format A4) = 'job '.
 - 2) AKEY (format A5)
Keyword defining mode: if 'xyz ' reads coordinates; if 'map ' reads electron density map.
 - 3) RATE LMINs LMINf CUTR STEP
RATE = defines the integration radius as $\text{RATE} \times \text{Molrad}$, where Molrad is radius of the smallest sphere, with origin at CoM, containing the whole molecule.
 $\left. \begin{array}{l} \text{LMINs} \\ \text{LMINf} \end{array} \right| = \left| \begin{array}{l} \text{several rotation functions are calculated, where the spherical-} \\ \text{harmonics expansions begin with } \text{LMINs} \leq \ell \leq \text{LMINf}; \text{ the} \\ \ell\text{-expansion controls the angular resolution.} \end{array} \right.$
CUTR = cutoff in rotation function output; first selects all peaks above $\text{CUTR} \times \text{maximum-peak-height}$.
STEP = step size for ϕ , θ and ψ (in degrees).
 - 4) TKEY NUMR CUTT NPIC (format A5,*)
 $\text{TKEY} = \left| \begin{array}{l} \text{'c-o'} \text{ computes centered-overlap;} \\ \text{'p-t'} \text{ computes phased-translation ('p-t-f' when phases} \\ \text{are available);} \\ \text{'h-l'} \text{ computes Harada-Lifchitz translation function;} \\ \text{'c-c'} \text{ computes correlation-coefficient.} \end{array} \right.$
NUMR = selects up to a maximum of NUMR orientations for input to one-body translations.
CUTT = cutoff in fast translation function output; first selects all peaks above $\text{CUTT} \times \text{maximum-peak-height}$ of 'c-o', 'p-t', 'h-l' or 'c-c' fast translation function.
NPIC = number of peaks to output of translation function. The program computes correlations and R-factors.
 - 5) Same as previous card, but for n-body translations (it must be present, even if not used).
 - 6) NUMT NITE
NUMT = selects up to a maximum of NUMT positions to refine.
NITE = number of iterations in the least-squares procedure.
 - 7) CUTD
Cutoff to eliminate positions with CoM-CoM distance less than CUTD Å.
-

For the above data, JOB proposes the following protocol (for a description of the different programs, see the write-up):

file ./job

```
# amore
./e/sorting
  set k=1
  while ({k} <= 2)
./e/tabling ${k}
  set k='expr ${k} + 1'
  end
  set k=1
  while ({k} <= 2)
./e/oic_rotting oicrd ${k} o${k}r
  set k='expr ${k} + 1'
  end
cat ./o/o*r.s > ./o/or1.s
./e/oic_traing oicto or1 ot1
  set k=1
  while ({k} < 4)
  set m='expr ${k} + 1'
./e/oic_traing oictn ot${k} ot${m}
  set k='expr ${k} + 1'
  end
./e/oic_fiting oicfd ot4 of4
./e/mr2ic of4 mr2ic
```

Outputs:

In the case of normal termination, directory ./o/ will then contain the following ascii files:

sort.s	=	statistics of the input reflections.
tabl1.s	=	information about the reference position of model 1.
tabl2.s	=	information about the reference position of model 2.
olr.s	=	cross-rotation peaks for model 1.
o2r.s	=	cross-rotation peaks for model 2.
or1.s	=	cross-rotation peaks for models 1 and 2.
ot1.s	=	one-body translation peaks for models 1 and 2.
ot2.s	=	two-body translation peaks for models 1 and 2 (with best solution in ot1.s kept fixed).
ot3.s	=	three-body translation peaks for models 1 and/or 2 (with best solution in ot2.s kept fixed).
ot4.s	=	four-body translation peaks for models 1 and/or 2 (with best solution in ot3.s kept fixed).
of4.s	=	fast rigid-body refinement of selected solutions in ot4.s .
mr2ic.s	=	rotation matrix and translation to apply to the input model in order to obtain the output coordinates.

The outputs of ROTING, TRAINING and FITTING follow a same pattern; the most general case is the n-body translation output:

file ./o/ot4.s = 4-body translation function (3 bodies kept fixed).

traing: ** D44HEL **

4	10									
> 1	103.0	26.2	12.7	0.0754	0.3067	0.4549	24.8	50.4	26.6	53.9
> 1	135.5	43.8	319.1	0.4601	0.2047	0.0151	33.0	48.0	37.6	41.5
> 2	107.2	70.6	15.8	0.4264	0.4266	0.6109	39.2	46.1	43.6	42.1
# 2	107.2	70.6	15.8	0.2885	0.0210	0.8054	37.0	46.7	41.9	93.0
# 2	107.2	70.6	15.8	0.3529	0.6612	0.9675	36.8	46.7	40.5	87.2
# 2	107.2	70.6	15.8	0.6897	0.7021	0.8633	36.7	46.6	40.9	91.3
# 2	107.2	70.6	15.8	0.5238	0.3561	0.7624	36.6	46.6	40.7	100.0
# 2	107.2	70.6	15.8	0.4997	0.3047	0.9067	36.6	46.5	40.1	97.3
# 2	107.2	70.6	15.8	0.1544	0.8305	0.0475	36.5	46.7	40.6	89.0
# 2	107.2	70.6	15.8	0.5273	0.1935	0.9461	36.5	46.9	41.1	83.5
# 2	107.2	70.6	15.8	0.4019	0.4385	0.7297	36.5	46.8	40.7	81.9
# 2	107.2	70.6	15.8	0.4901	0.3063	0.5681	36.4	46.6	41.7	95.0
# 2	107.2	70.6	15.8	0.4397	0.8707	0.6853	36.4	46.8	41.7	97.2
4	1									
> 1	103.0	26.2	12.7	0.0754	0.3067	0.4549	24.8	50.4	26.6	53.9
> 1	135.5	43.8	319.1	0.4601	0.2047	0.0151	33.0	48.0	37.6	41.5
> 2	107.2	70.6	15.8	0.4264	0.4266	0.6109	39.2	46.1	43.6	42.1
# 2	105.6	61.1	48.4	0.8012	0.5030	0.1681	42.1	44.9	47.3	98.0
4	10									

...

Description

- 1) Keyword (format A7) = 'traing:'.
- 2) NBOD NPIC
NBOD = number of molecules (n-body).
NPIC = number of translations for the given orientation.

Then NBOD-1 cards corresponding to the fixed positions:

- 3) $> \mu \phi \theta \psi x y z C_f R_f C_i D_m$
The last four items (descriptors) recall the genesis of the fixed position.
 μ = model identification number; also logical-unit identifier
for molecular scattering factors; usually, logical-unit = $\mu + 10$.
 ϕ, θ, ψ = Euler angles.
 x, y, z = translations (fractionary).
 C_f = correlation of amplitudes ($\times 100$).
 R_f = crystallographic R-factor ($\times 100$).
 C_i = correlation of intensities ($\times 100$).
 D_m = minimal CoM-CoM distance with current symmetry
related and preceding positions.

Then NPIC cards corresponding to the orientation that was translated:

- 4) $\# \mu \phi \theta \psi x y z C_f R_f C_i T_f$
The last three descriptors correspond to the whole configuration
(i.e. fixed positions plus translated orientation).
 μ = model identification number; also logical-unit identifier
for molecular scattering factors; usually, logical-unit = $\mu + 10$.
 ϕ, θ, ψ = Euler angles.
 x, y, z = translations (fractionary).
 C_f = correlation of amplitudes ($\times 100$).
 R_f = crystallographic R-factor ($\times 100$).
 C_i = correlation of intensities ($\times 100$).
 T_f = fast-translation-function value (% of maximum-value).

Repeat 2) to 4) for other positions.

The meaning of the descriptors in the output of ROTING is however different:

file ./o/o1r.s = cross-rotation function.

```
rotating: ** D44HEL **
  1      89
# 2   15.1  71.5  236.4  0.0000  0.0000  0.0000   9.2  54.2  15.5  11.5
# 2  107.2  70.6   15.8  0.0000  0.0000  0.0000  10.2  54.1  18.0  10.9
# 2   73.0  59.1   86.8  0.0000  0.0000  0.0000  10.0  54.1  19.3  10.2
# 2   25.5  82.8  242.3  0.0000  0.0000  0.0000   8.8  54.1  14.0  10.0
# 2  174.5  12.1   30.5  0.0000  0.0000  0.0000   8.8  54.4  13.3   9.9
...
```

Description

- 1) Keyword (format A7) = 'rotating:'.
- 2) NBOD NPIC
NBOD = number of molecules (always 1).
NPIC = number of orientations.

Then NPIC cards:

- 3) # μ ϕ θ ψ x y z C_f R_f C_i C_p
 μ = model identification number; also logical-unit identifier
for molecular scattering factors; usually, logical-unit = $\mu + 10$.
 ϕ, θ, ψ = Euler angles.
 x, y, z = translations (fractionary); set to zero.
 C_f = correlation of amplitudes, in P1 ($\times 100$).
 R_f = crystallographic R-factor ($\times 100$), in P1.
 C_i = correlation of intensities ($\times 100$). It includes all symmetry
related orientations, but no intermolecular contribution.
 C_p = truncated Patterson correlation ($\times 100$).
-

Errors

It is recommended to execute the programs SORTING and TABLING – the interface with the user – separately,

```
csh ./e/sorting  
csh ./e/tabling 1  
csh ./e/tabling {#}
```

Message errors appear in the log file or standard output.

- **Missing files:** The 'INPUT ERROR' means that a file-name passed as argument does not exist, or that the file does not match the option of the program called.
- **Dimension:** Errors give, if possible, explicit messages (e.g. “set mi >> ...”). Otherwise they just indicate the problem (e.g. “insufficient memory for ...”).

In all cases, fix dimensions in file ./e/maitre.for (FORTRAN code). This file contains the main calling program (amore) with a “parameter” card to define dimensions. Then

```
csh ./e/makeupd
```

it creates a new (local) executable (./e/amore.exe).

- **Format:** The FORMAT card of diffraction data and coordinates should be carefully checked (see Example above). Also, it is worth displaying the input model on a graphics system to check that the model does not contain atoms in extravagant positions. Occupation numbers are not used.
- **Searches with electron densities:** The overall strategy of automation is the fast – and reasonably accurate – computation of structure factors, which is possible if the TABLING program is correctly used (see the write-up). The user should be able to produce an electron density within a rectangular box of any desired dimension, at any desired resolution (within 15 % shift from theoretical requirements).